Algethami, Haneen and Landa-Silva, Dario (2015) A study of genetic operators for the Workforce Scheduling and Routing Problem. In: 11th Metaheuristics International Conference (MIC 2015), 7-10 June 2015, Agadir, Morocco.

A Study of Genetic Operators for the Workforce Scheduling and Routing Problem

Haneen Algethami[1], Dario Landa-Silva[2]

[1] University of Nottingham 1
School of Computer Science, Jubilee Campus, Nottingham, UK
hqa@notttingham.ac.uk

[2] University of Nottingham
School of Computer Science, Jubilee Campus, Nottingham, UK
dario.landasilva@nottingham.ac.uk

Abstract

The Workforce Scheduling and Routing Problem (WSRP) is concerned with planning visits of qualified workers to different locations to perform a set of tasks, while satisfying each task time-window plus additional requirements such as customer/workers preferences. This type of *mobile workforce* scheduling problem arises in many real-world operational scenarios. We investigate a set of genetic operators including problem-specific and well-known generic operators used in related problems. The aim is to conduct an in-depth analysis on their performance on this very constrained scheduling problem. In particular, we want to identify genetic operators that could help to minimise the violation of customer/workers preferences. We also develop two cost-based genetic operators tailored to the WSRP. A Steady State Genetic Algorithm (SSGA) is used in the study and experiments are conducted on a set of problem instances from a real-world Home Health Care scenario (HHC). The experimental analysis allows us to better understand how we can more effectively employ genetic operators to tackle WSRPs.

1 Introduction

The Workforce Scheduling and Routing Problem (WSRP) involves the scheduling of employees in an organization to travel across different locations to complete a set of tasks or activities. The problem arises in real-world scenarios such as Home Health Care, Home Care, Scheduling Technicians, Security Personnel Routing and Rostering and Manpower Allocation [3]. The WRSP is a combination of personnel scheduling and routing problems, hence it shares many aspects with these problems many of which are known to be NP-Hard [12]. The scheduling part allocates staff to tasks in order to fulfill work demand and customer/workers satisfaction. The routing part requires generating routes for *mobile staff* to service customers across various locations and within given time-windows. Real-world instances of the WSRP are large and very difficult to solve. It is necessary to develop efficient algorithms because manual planning involves a large effort and may produce low-quality plans.

Different approaches have been applied to WSRP, such as exact, heuristic and metaheuristic methods. A survey and a mathematical model were presented in [3]. In that work it was shown that a modern solver still takes considerable computational time to find solutions for instances with around 50 tasks and more. Therefore, a greedy heuristic for WSRP with time-dependent activities constraints was proposed more recently in [2]. That heuristic obtained better results in less computational time compared to the solver, especially in larger instances. Evolutionary Algorithms (EAs) are known to be capable of generating good-quality solutions in practical computational time. Genetic Algorithms (GAs) have been effective in finding good solutions relatively quickly to real-world scheduling problems like airline crew scheduling [10]. So far, few works have applied EAs to problems where scheduling and routing are combined. Examples include a multi-objective genetic algorithm (MOGA) [5], constructive search with variable neighbourhood search [17], particle swarm optimisation [1, 8, 13] and recently a group GA [15]. However, in those works attention was given to developing an algorithm to produce better solutions without explicitly investigating the genetic operators involved. Also, repair heuristics have been used to deal with infeasible solutions in such problems [1, 8]. The recent work in [15] shows that combining genetic and repair operators benefits the performance of EAs when tackling an example of WSRP.

In this paper we conduct a study of several genetic operators when applied to a WSRP using a Steady State Genetic Algorithm (SSGA) [19]. For this study we use a set of instances from a real-world home health care scenario provided by our industrial partner. Home health care involves providing services such as nursing, home life aids, physical therapy and house cleaning to patients at their homes. Section 2 describes the WSRP, model and instances used in this study. Section 3 gives details of the implemented genetic operators and GA. Section 4 describes the experimental study while Section 5 presents and discusses results. The paper is then concluded in Section 6.

2 Workforce Scheduling and Routing Problem (WSRP)

Each problem instance contains a set of locations $L = \{l_1, l_2, \ldots, l_{|L|}\}$ and each location $l \in L$ has a set of required tasks. Each task $t \in T$ has a time-window $[w_t^L, w_t^U]$ within which task must be serviced. A group of locations are assembled as a geographical area $a \in A$. Also, a problem instance has a set of workers $C = \{c_1, c_2, \ldots, c_{|C|}\}$. A worker $c \in C$ has a set of skills $S_c \subseteq S$ where $S = \{s_1, s_2, \ldots, s_{|S|}\}$ represents a set of all available skills. A worker $c \in C$ has to travel between locations in a graph $G = (V, E)$ [11] where the set of vertices V is the union of task locations, departure locations D, and destination locations D', that is $V = D \cup L \cup D'$. Then, the set of edges E is a set of routes between vertices in V. We have a number of instances from a real-world WSRP, specifically home health care scenario in the UK. The main features of the seven instances used in this work are shown in Table 1.

Instance	B-01	B-02	B-03	B-04	B-05	B-06	B-07
$\|C\|$	25	25	34	34	32	32	32
$\|L\|$	27	11	43	14	38	38	38
$\|T\|$	36	12	69	30	61	57	61
$\|A\|$	5	4	6	4	7	7	6

Table 1: Main Features of the WSRP Instances Used Here. $|C|$ is the number of workers, $|T|$ is the number of tasks, $|L|$ is the number of locations grouped in $|A|$ geographical areas.

Assigning tasks only according to workers' skills, satisfying time-windows and not exceeding weekly working hours are usually considered as hard constraints. Additional workers' requirements such as preferable areas to work or customers' requirements such as preference for certain workers to do the tasks are considered preferences hence tackled as soft constraints. The violation of soft constraints is penalized according to some established priority. In this study, the problem objectives and the set of constraints to be tackled are shown in Table 2. For more details on the MIP model please refer to [11]. In general, a WSRP solution is to assign each task to a worker while satisfying the constraints and minimizing distance travelled and cost. Hence, the objective function in equation (1) includes the **Assignment Cost** and the **Penalty Cost**. The Assignment Cost includes *Payment Cost and Travelling Cost*, i.e. wages plus the journeys cost for each worker. The Penalty Cost is the accumulated penalty for the violations of the soft constraints listed in Table 2.

Objectives	**Hard constraints**	**Soft constraints**
Distance Travelled	Skills Requirements	Unassigned Tasks
Payment Costs	Time Conflicts	Area Availability
	Maximum Hours Violations	Time Availability
		Preferences (Skills, Workers, Area)

Table 2: Objectives and Constraints in the WSRP.

$$\text{Min} \sum_{c \in C} \sum_{i \in D \cup T} \sum_{j \in D' \cup T} \left(d_{i,j} + p_j^c\right) x_{i,j}^c + \sum_{c \in C} \sum_{i \in D \cup T} \sum_{j \in D' \cup T} \rho_j^c x_{i,j}^c + \sum_{j \in T} M_1 y_j + M_2(\omega_j + \psi_j) \quad (1)$$

where:

- Binary decision variable $x_{i,j}^c = 1$ if worker $c \in C$ is assigned to task $j \in T$ after servicing task $i \in T$, $x_{i,j}^c = 0$ otherwise.
- y_j is a binary decision variable indicates that a task $j \in T$ is assigned to a dummy worker. An assignment only takes place when the task j cannot be assigned to a real worker due to the problem restrictions.
- M_1 and M_2 are penalty costs applied to unassigned tasks and constraints violation penalties. The values are provided along with data instances.
- The Payment Cost is denoted by p_j^c for worker $c \in C$ to perform task $j \in T$
- The Travelling Cost is denoted by $d_{i,j}$ from location of task $i \in T$ to the location of task $j \in T$.
- The Penalty Cost due to the violation of soft constraints is denoted by ρ_j^c.

Priority among the six soft constraints is as follows. The *Unassigned Tasks* constraint has the highest priority so a violation of this constraint costs more than a violation of any of the other five constraints. Both *Time Availability* and *Area Availability* constraints have the same priority. Note that there are two constraints associated to the area in which workers are assigned to tasks. If a worker is assigned outside his/her *Area Availability* then this involves a constraint penalty cost. Likewise, if a worker is not assigned to one of his/her *Preferred Area* then this involves a preference penalty cost. Also, *The worker's and client's preferences* ρ_j^c is a penalty cost that varies according to the preference required. We introduce three preference requirement levels for each preference constraints in the problem. The first requirement level is the preferred worker qualifications R_{L_s}, then the preferred area to work R_{L_a}, and finally the preferred worker R_{L_c}. Such preference satisfaction level is in the range $[0, 1]$ depending on the level of preference achieved.

3 Genetic Algorithm, Conflict Reduction Heuristic and Genetic Operators

This work aims to study the effect of various genetic operators within a Steady State Genetic Algorithms (SSGA) for the WSRP. Such relatively not elaborate genetic algorithm [19] was selected for our study in order to analyse the emergent behaviour and performance of the genetic operators working on a relatively straightforward GA implementation. Using a straightforward solution representation allows to apply the genetic operators directly on the phenotype [18]. Thus, a simple *direct representation scheme* is used for chromosome encoding. A vector of integers of length $|T|$ tasks represents a day plan and each integer holds the information of a worker assigned to the corresponding task. To increase the chance of generating an initial feasible solution, integers in the chromosome vector are sorted in increasing order according to the task start times so that we can determine which task is served first [15]. Of course the same worker can be assigned to more than one task and not all workers might be used in the day plan. An initial population of m individuals is randomly generated each representing a daily plan. We aim to generate new feasible offspring that have lower total cost per worker. Often a random population contains daily plan with overlapping time-windows for different tasks per worker, called time-conflicts. Hence, we require a heuristic to re-assign tasks and hence reduce such time-conflicts. A Time-window Conflict Reduction Heuristic (TCRH) is applied to the random generated individuals.

This TCRH aims to seed the GA with better initial solutions by reducing the number of time-conflicts generated with the random process. Basically, the heuristic re-assigns time-conflicting tasks to a different random worker. Tournament selection and elitism is used at the end of each generation so that 33.3% of the best individuals are copied to the new population. Individuals are ranked according to their fitness given by the minimization objective function.

Operator	RSM	IM	SM	1-point	2-point	UX	HX	PMX	OX	CX
Reference	[4]	[7]	[4]	[14]	[9]	[14]	[6]	[21]	[20]	[16]

Table 3: Operators considered in the study and adopted from the literature.

Various genetic operators, including crossover and mutation, are applied to the chromosomes in order to obtain better offspring. In this study, there are three different types of operators: ten well-known generic operators (shown in Table 3), two cost-based problem specific operators designed to obtain offspring with fewer violations, and a repair operator. These operators were selected after a literature survey conducted to identify genetic operators that have been applied to WSRP and related problems. By studying the overall performance of these operators we seek to identify the best combination of operators for tackling the WSRP constraints. Each operator allows feasible solutions only. As a result, the operators always create offspring with fewer constraint violations regardless of the total cost. The operators used in the study are as follows:

- **Well-known Generic Operators**
  - Crossover These well-known generic crossovers are divided in two groups:
    1. Scheduling crossovers such as: Single Point (1-point), Two Point (2-point), Uniform (UX) and Half Uniform (HX).
    2. Routing crossovers such as: Order crossover (OX), Cycle crossover (CX) and partially matched crossover (PMX).
  - Mutation Applied mutation operators are: Random Swap Mutation (RSM), Inversion Mutation (IM) and Scramble Mutation (SM).

Algorithm 1 Cost-Based Uniform Crossover (CBUX).

Selects k in Individual i_{Parent}
if Worker c is Available to work in Task t_k area in i_{Parent} **then**
  Worker c is Inserted into the k^{th} gene in i_{Child}
end if
Evaluate the *Cost Value* i_{Child}
if i_{Child} has more Violation than Parent i_{Parent} **then**
  $i_{Child} := i_{Parent}$
end if

- **Cost-Based Operators**
  These are problem-specific operators that are specifically designed for WSRP to maximize the customer/workers preferences satisfaction.
  - Cost-Based Uniform Crossover (CBUX).
    In the study on the real-world airline crew scheduling problem [10], the proposed cost-based crossover produced good results by restricting the mating pool to feasible solutions only and limiting the search to the nearest task available for each worker at his current location. Hence, we developed a Cost-Based Uniform Crossover (CBUX) operator that selects a gene based on the preferred time availability of a worker to work on a task as illustrated in Algorithm 1. Genes are selected according to a cost instead of using a uniform mask by maintaining a

stream of workers in an offspring provided that it does not make the chromosome infeasible. Therefore, the selected gene has lower cost.

– Cost-Based Mutation (CBM).

This operator is designed to target each gene in an individual by maintaining the maximum number of preferred workers for each task as shown in Algorithm 2. Each Task $t_i \in T$ has a list of preferred workers $C = [c_1, c_2 \ldots, c_M]$ where M is the number of workers in the preferred list. The CBM operator checks the preferred workers and alters the genes accordingly.

- **Repair Operator**
  The feasibility of an individual is gone if a hard constraint is violated. Therefore, the repair operator is designed to maintain feasible solutions while maintaining the least possible cost value. The operator checks the feasibility of each worker assigned to a task and alters the genes accordingly. This procedure is designed to reach a good area in the search space in a fast computation time. At every generation, the repair operator is utilized after the mutation. A random swap occurs if a worker is not suitable for a task, provided that the alteration does not create infeasible solutions.

Algorithm 2 Cost-Cased Uniform Mutation (CBM).

```
for all Gene k in Individual i_Parent do
    if Worker c is in a Preferred Worker for Task t_k in i_Parent then
        Worker c is Inserted into the kth gene in i_Child
    end if
end for
Evaluate the Cost Value i_Child
if i_Child has more Violation than Parent i_Parent then
    i_Child := i_Parent
end if
```

4 Experimental Study

The aim of the experiments was to investigate the effect of the set of genetic operators listed above on the worker's and customer's preferences satisfaction. The experiments were conducted on 3.60 GHz Intel (R) core (TM) i7 CPU and the implementation was in Java under Windows 7. The experimental setting was as follows. The population of a size 100 was generated randomly with initial seed yet initial individuals were improved with the TCRH. The crossover probability p_c was alternated between 100% , 50% and 10%. Similarly the mutation probability p_m 30% , 10% and 0.1%, i.e. we tried two different values for crossover and mutation probabilities. Each GA setting was executed using different combinations of genetic operators on each of the seven problem instances mentioned in Table 1. In total we had 32 combination with 9 different settings of the GA alternately and each one was executed 15 times with the same initial population. Each run stops at the generation stagnation (with a maximum number of 200 iterations). For each GA setting, we kept the objectives value for the best solution obtained in each of the 15 runs and the average of those best values is used to measure the quality of the GA setting. Also, a repair operator was executed to reduce the number of soft constraints violations while maintaining a flow of feasible solutions. Consequently, we grouped each run by the mutation operators where RSM, IM, SM and CBM. Each mutation operator was combined with one crossover operator (1-point, 2-point, UX, HX, PMX, OX, CX, CBUX) at a time. However, to investigate the behaviour of the repair operator, the approach was tested with two additional settings. If a combination was not combined with a repair operator, then the combinations were denoted as X^*. Then the combinations with the repair operator and probabilities of p_c= 0.1% and p_m = 0.01% (i.e. practically no crossover and no mutation). These values represent the combinations that test very low genetic operator probabilities.

5 Results and Discussion

Having time-conflicts in the generated solutions, i.e. the violation of time-windows, was a problematic issue in this study. TCRH was implemented to reduce the number of such time-conflicts. The quality of the random initial population and the TCRH methods are compared in Figure 1. TCRH performance is directly measured by two elements: the number of time-conflicts and the initial population quality. Hence, in in Figure 1, each bar presents the average value of the time-window conflicts occurrences for each problem instance before and after applying TCRH. It can be seen that after applying TCRH there is a clear reduce in time-window conflicts for all instances. The second measurement method is the population quality. This measurement is reflected by the population diversity and the computation time. As shown in Table 4, the computation time increases when TCRH is applied. However, instances such as B-01 and B-02 required less time while using TCRH and B-04 required similar computation time to the random initialisation. This outcome was a result of few time-window conflicts primarily even with the random initialization. Furthermore, the population diversity is equally the same in all instances for both initialisation methods. Thus, population diversity is maintained while TCRH is used with the GA. In general, we observed that TCRH reduced the number of workers with time-conflict occurrences in each problem instance with the utilization of the repair operator.

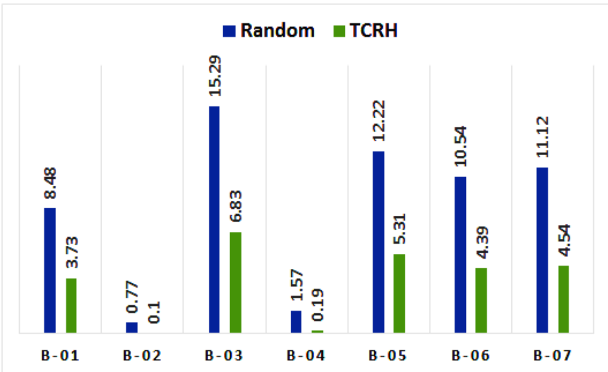

Figure 1: The Average Number of Conflicts in the Initial Population P (Random / TCRH).

	Random							TCRH						
Problem B	**01**	**02**	**03**	**04**	**05**	**06**	**07**	**01**	**02**	**03**	**04**	**05**	**06**	**07**
$time(ms)$	83	12	1	0	16	34	18	31	0	62	0	31	15	27
$Diversity(\%)$	24	24	31	31	30	30	30	24	24	31	32	30	30	30

Table 4: Effects of the TCRH on the initial Population P.

For the effect of studying the performance of the various genetic operators on maximizing customer's and workers preferences is stated in Table 5. The table shows the wilcoxon sign rank test for the mean preference satisfaction values for each problem instance. The wilcoxon sign rank test is non-parametric statistical hypothesis test used to compare the average of two groups of values of the preferences satisfaction to evaluate whether their population mean ranks differ. The statistical comparison is applied two crossover operators at a time to provide an overall evaluation of the method. We can see crossover operators such as CX and PMX were the least effective. Generally, CBUX performed the best on instances B-01, B-05, B-06 and B-07. Another observation that we made from our detailed experimental results is that CBUX obtained less constraint violations in 35% over all the experiments. Nevertheless, k-point crossover operators obtained 14% of all the minimum cost values. This shows that these generic k-point crossover operators are not always the best to be used in the WSRP. Table 5 reports for each problem instance, results from using different probabilities crossover and mutation probabilities (δ, γ) with the different combination of genetic operators. The columns of the Table give the mean cost value

B-01

	1Point	2Point	UX	PMX	OX	HX	CX	CBUX
1Point	-	$<$	$<$	$>$	$<$	$>$	$<$	$<$
2Point	$>$	-	$>$	$>$	$>$	$>$	$>$	$<$
UX	$>$	$<$	-	$>$	$<$	$>$	$<$	$<$
PMX	$<$	$<$	$<$	-	$<$	$>$	$<$	$<$
OX	$>$	$<$	$>$	$>$	-	$>$	$>$	$<$
HX	$<$	$<$	$<$	$<$	$<$	-	$<$	$<$
CX	$>$	$<$	$>$	$>$	$<$	$>$	-	$<$
CBUX	$>$	$<$	$>$	$>$	$>$	$>$	$>$	-

B-02

	1Point	2Point	UX	PMX	OX	HX	CX	CBUX
1Point	-	$>$	$>$	$>$	$>$	$>$	$>$	$>$
2Point	$<$	-	$>$	$>$	$>$	$>$	$<$	$<$
UX	$<$	$<$	-	$<$	$<$	$>$	$<$	$<$
PMX	$<$	$<$	$>$	-	$<$	$<$	$<$	$<$
OX	$<$	$<$	$>$	$>$	-	$>$	$<$	$<$
HX	$<$	$<$	$<$	$>$	$<$	-	$<$	$<$
CX	$<$	$>$	$>$	$>$	$>$	$>$	-	$<$
CBUX	$<$	$>$	$>$	$>$	$>$	$>$	$>$	-

B-03

	1Point	2Point	UX	PMX	OX	HX	CX	CBUX
1Point	-	$<$	$\leq$	$\leq$	$\leq$	$\leq$	$\leq$	$\leq$
2Point	$>$	-	$<$	$>$	$>$	$<$	$<$	$<$
UX	$\geq$	$>$	-	$>$	$>$	$>$	$<$	$<$
PMX	$\geq$	$<$	$<$	-	$<$	$<$	$\leq$	$\leq$
OX	$\geq$	$<$	$<$	$>$	-	$<$	$<$	$<$
HX	$\geq$	$>$	$<$	$>$	$>$	-	$<$	$<$
CX	$\geq$	$>$	$>$	$\geq$	$>$	$>$	-	$<$
CBUX	$\geq$	$>$	$>$	$\geq$	$>$	$>$	$>$	-

B-04

	1Point	2Point	UX	PMX	OX	HX	CX	CBUX
1Point	-	$<$	$<$	$<$	$>$	$<$	$>$	$<$
2Point	$>$	-	$<$	$>$	$>$	$>$	$>$	$>$
UX	$>$	$>$	-	$>$	$\geq$	$>$	$>$	$>$
PMX	$>$	$<$	$<$	-	$>$	$<$	$>$	$<$
OX	$<$	$<$	$\leq$	$<$	-	$<$	$<$	$<$
HX	$>$	$<$	$<$	$>$	$>$	-	$>$	$>$
CX	$<$	$<$	$<$	$<$	$>$	$<$	-	$<$
CBUX	$>$	$<$	$<$	$>$	$>$	$<$	$>$	-

B-05

	1Point	2Point	UX	PMX	OX	HX	CX	CBUX
1Point	-	$<$	$<$	$<$	$<$	$<$	$<$	$\leq$
2Point	$>$	-	$<$	$<$	$>$	$>$	$<$	$\leq$
UX	$>$	$>$	-	$>$	$>$	$>$	$<$	$<$
PMX	$>$	$>$	$<$	-	$<$	$>$	$<$	$<$
OX	$>$	$<$	$<$	$>$	-	$>$	$<$	$<$
HX	$>$	$<$	$<$	$<$	$>$	-	$<$	$<$
CX	$>$	$>$	$>$	$>$	$>$	$>$	-	$<$
CBUX	$\geq$	$\geq$	$>$	$>$	$<$	$>$	$>$	-

B-06

	1Point	2Point	UX	PMX	OX	HX	CX	CBUX
1Point	-	$<$	$<$	$<$	$<$	$>$	$>$	$<$
2Point	$>$	-	$>$	$>$	$>$	$>$	$>$	$<$
UX	$>$	$<$	-	$<$	$<$	$>$	$>$	$<$
PMX	$>$	$<$	$>$	-	$>$	$>$	$>$	$<$
OX	$>$	$<$	$>$	$<$	-	$>$	$>$	$<$
HX	$<$	$<$	$<$	$<$	$>$	-	$>$	$<$
CX	$<$	$<$	$<$	$<$	$>$	$<$	-	$<$
CBUX	$>$	$>$	$>$	$>$	$>$	$>$	$>$	-

B-07

	1Point	2Point	UX	PMX	OX	HX	CX	CBUX
1Point	-	$<$	$<$	$<$	$>$	$>$	$>$	$<$
2Point	$>$	-	$>$	$<$	$>$	$>$	$>$	$<$
UX	$>$	$<$	-	$<$	$>$	$>$	$>$	$<$
PMX	$>$	$>$	$>$	-	$>$	$>$	$>$	$<$
OX	$<$	$<$	$<$	$<$	-	$>$	$>$	$<$
HX	$<$	$<$	$<$	$<$	$>$	-	$<$	$<$
CX	$<$	$<$	$<$	$<$	$>$	$>$	-	$<$
CBUX	$>$	$>$	$>$	$>$	$<$	$>$	$>$	-

Table 5: Wilcoxon sign rank test of the Average Preferences Satisfaction Values Grouped by the Crossover Operators where $>(<)$ means that Crossover i is significantly better (worse) than Crossover j within a confidence interval of 95%. Similarly, $\geq$ $(\leq)$ shows that Crossover i performs slightly better than Crossover j (with no statistical significance). The sign = refers to equal performance.

μ, standard deviation σ, the best values obtained are highlighted in bold and the average computation time in seconds used by the GA. It is apparent that the combinations X^* were not affected by the genetic operators for all instances. They sustained similar outcomes throughout the trials with a near to random offspring. Overall, CBM had less spread out values with lower σ values in association to the other combinations. Additionally, CBM have succeeded to obtain the best cost values in B-01 and B-7. RSM achieved best cost values as well in both B-02 and B-03. Both obtained less σ values throughout all the instances. On the contrary, IM obtained best cost values in B-04, B-05 and B-06. Yet IM had more computation time and spread out values overall the instances as well. For the most part, lower P_c, P_m values have obtained good solutions in less computation time in comparison with other combinations settings for all instances excluding B-05. In the cases of B-01 and B-02, best obtained value is achieved when high P_m is used. However, low probability preformed worse in B-05 with Rsm and IM in particular while and best obtained values when high P_c is used. On the other hand, the optimal cost values for each datasets by mathematical programming solver are 1.7, 1.8, 1.7, 2.07, 1.8, 1.6 and 1.79 constitutively with a computation time 21, 2, 6003, 25, 585, 184 and 300 in seconds. Despite that the cost values are better than the GA, it is clear that GA obtained good results in less computation time throughout all the setting tested. Mainly, the experiments goal is to evaluate suitable combinations of operators that can perform well under a GA for WSRP and later embedded in a more efficient approach.

			RSM				IM				SM				CBM			
Instance	δ	γ	σ	μ	min	$time(s)$	σ	μ	min	$time(s)$	σ	μ	min	$time(s)$	σ	μ	min	$time(s)$
B-01	X^*		25.39	74.37	40.96	9.72	25.39	74.37	40.96	9.72	25.39	74.37	40.96	9.72	25.39	74.37	40.96	9.72
	1.0	0.001	0.02	2.4	2.35	18.75	0.04	2.42	2.35	15.38	0.04	2.42	2.35	15.65	0.03	2.41	2.35	12
	1.0	0.01	0.08	2.44	2.34	17.38	0.08	2.44	2.34	16.75	0.1	2.45	2.34	14.25	0.05	2.4	2.34	13.43
	1.0	0.1	0.08	2.39	**2.28**	18.63	0.07	2.39	**2.28**	24.5	0.06	2.36	2.28	18.75	0.07	2.36	2.28	16.43
	1.0	0.3	0.05	2.45	2.37	15.63	0.07	2.45	2.37	26.75	0.05	2.4	2.31	17.25	56.7	25.61	2.38	16.29
	0.5	0.001	0.05	2.42	2.32	16.63	0.06	2.43	2.31	19.75	0.04	2.44	2.39	17.13	0.03	2.44	2.39	15.14
	0.5	0.01	0.06	2.4	2.3	17.63	0.07	2.44	2.33	19.63	0.06	2.49	2.38	17	0.06	2.41	2.31	14
	0.5	0.1	0.04	2.41	2.35	21	0.06	2.46	2.39	29.25	0.07	2.41	2.3	20.88	0.05	2.44	2.35	20.14
	0.5	0.3	0.08	2.44	2.34	20.88	0.05	2.39	2.31	30	0.03	2.42	2.35	21.5	0.05	2.42	2.35	18.14
	0.1	0.001	0.05	2.43	2.38	9.38	0.05	2.44	2.38	9.63	53.55	22.65	2.36	9.38	0.05	2.42	2.35	7.86
	0.1	0.01	0.07	2.43	2.31	11	0.08	2.4	2.3	14.5	0.08	2.41	**2.27**	10.63	0.08	2.41	2.29	8.71
	0.1	0.1	0.06	2.44	2.35	12	0.07	2.42	2.32	23.88	0.07	2.45	2.32	15.38	0.07	2.4	2.3	10.43
	0.1	0.3	0.07	2.43	2.34	12.88	0.05	2.41	2.34	24.88	0.05	2.43	2.38	12.88	0.04	2.42	2.38	11.43
	0.01	0.001	0.05	2.44	2.36	0.13	0.04	2.43	2.36	0.13	0.07	2.42	2.32	0.13	0.05	2.46	2.36	0.43
	0.01	0.01	0.08	2.45	2.32	0.38	162.13	83.49	2.37	2.38	0.06	2.37	2.24	0.5	0.08	2.4	2.28	0.57
	0.01	0.1	0.05	2.4	2.34	0.38	53.59	22.71	2.4	23	214.26	83.45	2.41	0.38	0.12	2.38	**2.15**	0.14
	0.01	0.3	0.04	2.42	2.36	0.38	53.57	22.69	2.38	25.63	0.07	2.42	2.32	0.38	0.06	2.43	2.34	0.57
B-02	$X*$		0.46	4.11	3.31	3.72	4.19	9.38	2.67	7.24	0.62	3.9	2.82	2.43	0.47	4.13	3.08	2.29
	1.0	0.001	0.12	2.46	2.26	2.63	0.12	2.41	2.25	3	0.05	2.44	2.37	2.75	0.14	2.43	2.18	1.25
	1.0	0.01	0.08	2.4	2.28	2.5	0.12	2.41	2.23	3.25	0.16	2.45	**2.13**	2.63	0.13	2.47	2.2	3
	1.0	0.1	0.09	2.53	2.35	2.75	0.16	2.48	2.22	7.88	0.16	2.38	2.19	2.5	0.1	2.49	2.39	3
	1.0	0.3	0.09	2.4	2.27	2.38	0.09	2.43	2.27	8.75	0.12	2.41	2.17	3.13	0.1	2.4	2.17	3
	0.5	0.001	0.11	2.43	2.29	2.88	0.13	2.52	2.31	2.75	0.16	2.43	2.25	2.63	0.05	2.49	2.42	1.86
	0.5	0.01	0.08	2.51	2.39	2.88	0.15	2.42	2.22	4.13	0.14	2.49	2.29	2.75	0.1	2.52	2.3	3
	0.5	0.1	0.12	2.48	2.2	2.75	0.06	2.49	2.41	8.88	0.15	2.46	2.26	2.63	0.14	2.39	**2.16**	2.71
	0.5	0.3	0.12	2.47	2.23	2.63	0.13	2.54	2.35	9	0.09	2.48	2.3	2.75	0.06	2.49	2.42	2.86
	0.1	0.001	0.12	2.4	2.16	1.63	0.14	2.42	2.19	1.75	0.2	2.51	**2.13**	1.63	0.1	2.44	2.24	1
	0.1	0.01	0.11	2.46	2.33	1.63	0.11	2.52	2.34	2.75	0.1	2.46	2.26	1.63	0.07	2.47	2.33	1.86
	0.1	0.1	0.08	2.43	2.34	1.63	0.15	2.48	2.24	8.13	0.13	2.46	2.23	1.63	0.14	2.49	2.34	1.86
	0.1	0.3	0.08	2.48	2.37	1.63	0.09	2.44	2.3	8.13	0.16	2.45	2.14	1.63	0.08	2.49	2.37	1.86
	0.01	0.001	0.14	2.4	2.13	0	5.9	4.85	2.44	0	6.16	4.77	2.38	0	0.13	2.44	2.25	0
	0.01	0.01	0.41	2.58	2.3	0	5.87	4.99	2.4	0.25	0.11	2.45	2.22	0	0.16	2.45	2.14	0
	0.01	0.1	0.11	2.5	2.27	0.75	0.15	2.43	2.23	5.88	0.1	2.42	2.33	0	0.04	2.51	2.45	0
	0.01	0.3	0.12	2.37	**2.12**	0.75	0.13	2.43	**2.15**	6.13	6.26	4.87	2.4	0	0.13	2.42	2.22	0
B-03	$X*$		1261.05	1753.87	17.38	435.88	1261.05	1753.87	17.38	435.88	471.08	1148.68	20.26	31.38	499.67	1073.95	19.34	291.63
	1.0	0.001	795.28	380	2.53	43.38	95.28	380	2.53	43.38	292.36	229.06	2.48	42.13	584.67	455.46	2.56	42.88
	1.0	0.01	523.16	304.62	2.48	49.75	523.16	304.62	2.48	49.75	795.26	380.01	2.55	43.25	261.59	153.65	**2.51**	43.63
	1.0	0.1	420.4	229.09	2.52	80.38	420.4	229.09	2.52	80.38	199.65	78.14	2.56	40.25	399.53	455.55	2.54	51.63
	1.0	0.3	292.37	229.06	2.51	86.63	292.37	229.06	2.51	86.63	795.42	380.09	2.6	43	420.38	229.1	2.52	91.88
	0.5	0.001	0.05	2.42	**2.32**	16.63	199.68	78.07	2.54	53	199.7	78.06	2.52	54.63	199.68	78.07	2.54	48
	0.5	0.01	798.86	304.55	2.55	60.38	798.86	304.55	2.55	60.38	0.06	2.57	**2.46**	53.5	0.04	2.59	2.55	51.25
	0.5	0.1	766.3	531.19	2.56	85.63	766.3	531.19	2.56	85.63	0.04	2.57	2.54	54.13	0.04	2.58	2.52	52.75
	0.5	0.3	798.89	606.67	2.58	85.63	798.89	606.67	2.58	85.63	0.05	2.6	2.51	49.75	0.16	2.63	2.51	55.88
	0.1	0.001	0.05	2.43	2.38	9.38	584.86	455.61	**2.5**	27.38	261.59	153.62	2.55	26.13	261.5	455.47	2.54	28.63
	0.1	0.01	823.5	531.05	2.57	42.5	823.5	531.05	2.57	42.5	199.85	78.15	2.56	27.13	877.2	682.05	2.56	28.13
	0.1	0.1	675.21	606.79	2.59	80.63	675.21	606.79	2.59	80.63	427.14	304.67	2.55	27.75	399.42	153.57	2.56	26.63
	0.1	0.3	1119.87	1059.57	2.56	82	1119.87	1059.57	2.56	82	604.2	304.71	2.53	27.5	658.18	455.54	2.54	29.63
	0.01	0.001	784.87	455.75	2.55	8.25	784.87	455.75	2.55	8.25	854.16	606.68	2.55	2.88	724.18	757.61	2.57	11.38
	0.01	0.01	420.28	379.98	2.48	11.5	1279.47	1588.46	2.59	11.5	840.72	757.71	2.54	9.38	302.14	304.75	2.59	4
	0.01	0.1	471.74	531.24	2.59	47.63	517.4	833.26	2.65	47.63	798.84	606.9	2.55	3.38	292.52	229.21	2.55	7.13
	0.01	0.3	658.69	455.87	2.53	49.63	517.57	380.1	2.58	49.63	795.39	380.19	2.53	3.5	500.82	455.56	2.54	9
B-04	$X*$		0.9	9.85	8.57	9	96.14	79.95	6.7	9.13	1.49	10.01	6.67	398.25	1.56	9.71	6.78	9.13
	1.0	0.001	0.07	2.65	2.6	12.63	0.06	2.69	2.57	11.88	0.06	2.71	2.61	12.38	0.09	2.72	2.56	12.38
	1.0	0.01	0.18	2.75	2.59	12	0.07	2.65	2.55	12.5	37.22	16.74	2.56	10.72	0.08	2.7	2.57	11.5
	1.0	0.1	0.07	2.71	2.59	11.5	0.04	2.7	2.63	24.25	0.08	2.69	2.54	13.89	0.1	2.7	2.53	11.75
	1.0	0.3	0.07	2.7	2.56	11.75	0.07	2.65	2.54	26.63	0.22	2.74	2.59	14.7	0.07	2.69	2.59	11.63
	0.5	0.001	0.05	2.73	2.64	14.13	0.1	2.67	**2.49**	14.63	0.04	2.75	2.68	14.63	0.09	2.68	**2.53**	13.5
	0.5	0.01	0.07	2.69	2.61	13.25	0.07	2.72	2.57	17.75	0.08	2.74	2.57	11.69	0.1	2.74	2.55	14.25
	0.5	0.1	0.12	2.71	2.55	14.25	0.04	2.68	2.59	26.13	0.05	2.73	2.65	15.45	0.07	2.71	**2.53**	13.88
	0.5	0.3	0.08	2.68	**2.52**	11.88	0.08	2.68	2.57	26.63	0.06	2.71	2.59	15	0.09	2.72	2.58	13.38
	0.1	0.001	0.08	2.7	2.6	5.75	0.06	2.68	2.55	6.13	0.07	2.71	2.61	5.38	0.06	2.71	2.6	5.88
	0.1	0.01	0.09	2.69	2.53	6	0.18	2.73	2.51	14.5	0.1	2.75	2.6	7.24	0.08	2.69	2.58	5.38
	0.1	0.1	0.1	2.73	2.54	5.38	37.68	16.96	2.57	24.5	0.07	2.7	2.56	10.41	0.11	2.73	**2.53**	5.63
	0.1	0.3	0.06	2.7	2.64	5.25	48.94	30.92	2.52	24.88	0.08	2.7	**2.53**	10.65	0.23	2.78	2.55	5.5
	0.01	0.001	0.07	2.73	2.63	0.38	74.72	31.02	2.64	0.63	0.07	2.68	2.59	0.5	0.07	2.72	2.63	0.25
	0.01	0.01	0.07	2.66	2.56	0.25	54.91	45.31	2.6	1.25	0.095	2.70	2.6	0.38	0.06	2.72	2.64	0.38
	0.01	0.1	0.43	2.95	2.59	0.25	0.21	2.73	2.51	18.25	0.06	2.71	2.61	0.25	0.2	2.8	2.61	0.5
	0.01	0.3	0.08	2.72	2.59	0.25	74.61	31.21	2.57	20.38	0.07	2.71	2.59	0.25	0.07	2.69	2.58	0.25

			RSM				IM				SM				CBM			
Instance	δ	γ	σ	μ	min	$time(s)$	σ	μ	min	$time(s)$	σ	μ	min	$time(s)$	σ	μ	min	$time(s)$
B-05	$X*$		464.53	378.66	17.59	27.25	623.89	1100.34	499.06	66	158.9	558.29	496.08	26.13	374.43	440.43	19.06	25.88
	1.0	0.001	788.19	844.22	2.53	47.13	1457.02	1564.95	2.5	57.5	1163.74	1384.96	2.52	54.38	1040.78	603.67	2.52	46.75
	1.0	0.01	1236.16	1505.26	**2.43**	56	677.68	663.79	**2.39**	407.75	1235.58	903.71	2.55	88.57	1018.17	1625.29	2.58	339
	1.0	0.1	1266.35	1084.26	2.53	70.63	1127.71	1204.99	2.48	557.13	1333.23	844.25	2.52	640.88	923.2	603.52	2.55	804
	1.0	0.3	1401.34	1925.45	2.59	397.13	858.21	1565.55	484.6	441.63	1421.77	1445.07	2.6	223.75	698.11	904.72	2.57	260.88
	0.5	0.001	560.35	543.2	2.5	49	1219.89	1084.32	2.55	53.38	813.22	543.88	2.51	59.5	795.24	784.01	2.54	53.38
	0.5	0.01	983.47	603.33	2.49	67.63	1125.49	663.47	2.54	253.5	1188.21	903.97	2.577	77.13	1068.01	603.77	**2.47**	73.75
	0.5	0.1	1177.05	723.36	2.46	265.13	830.48	1145.31	2.6	70.75	588.77	723.75	**2.45**	50.38	1482.08	1204.68	2.49	441.75
	0.5	0.3	1113.03	904.1	2.56	937	1225.69	1685.7	2.48	70.75	847.82	904.11	2.51	619.38	991.04	964.29	2.57	62.88
	0.1	0.001	1099.79	1264.73	2.57	14.63	1120.91	1565.03	2.6	24.38	823.94	603.53	2.54	22.25	610.09	904.27	2.52	25.25
	0.1	0.01	1611.86	1925.75	2.67	26.63	1195.87	1324.63	2.5	41.25	1019.74	1204.51	2.48	26.38	506.73	904.44	2.58	283
	0.1	0.1	1437.37	1324.76	2.55	25.13	944.14	1865.99	2.64	281.38	824.3	1084.88	2.47	61	585.65	663.65	2.48	26.63
	0.1	0.3	922.8	1325.15	2.62	105	1368.63	2346.65	2.51	243.25	1325.71	1384.72	2.59	24.13	718.31	784.03	2.66	82.5
	0.01	0.001	824.17	2046.45	964.6	14.25	721.71	2406.84	963.95	11.13	1046.61	1265	2.58	7.25	1139.1	1505.33	2.56	3.5
	0.01	0.01	1048.11	1685.95	2.67	9.25	797.12	1925.83	2.55	6.13	1593.04	2106.07	2.47	4.5	1534.56	2046.4	2.64	4.25
	0.01	0.1	1113.04	1985.92	2.59	2.88	1175.73	2226.28	483.55	49.88	1294.12	1685.39	2.6	7.5	1068.26	1564.97	2.62	5.63
	0.01	0.3	1020.05	1926.13	2.53	3.13	711.14	2286.73	1444.57	53.5	1127.85	1445.65	2.59	3.38	1032.38	1865.69	483.56	13.63
B-06	$X*$		186.09	254.79	12.37	60	694.44	593.74	14.85	60	166.8	110.6	12.95	23.75	254.87	303.08	12	24.38
	1.0	0.001	166.78	98.8	2.46	28.5	127.48	50.81	2.41	28.88	254.74	98.79	2.43	28.88	268.05	146.98	2.5	29.25
	1.0	0.01	280.65	222.62	2.5	34.75	385.43	195.23	2.45	34.38	319.52	291.47	2.5	28.7	536.12	291.4	**2.4**	30
	1.0	0.1	134.95	57.6	**2.39**	28.88	127.37	50.68	2.5 5	8.63	127.54	50.71	**2.39**	34.88	268.17	147.02	2.47	29.38
	1.0	0.3	673.81	497.66	2.45	36.13	0.36	2.62	2.44	64	651.14	435.89	2.45	38.52	166.86	98.89	2.47	36
	0.5	0.001	0.06	2.49	2.4	34.5	0.04	2.52	2.43	34.13	0.07	2.49	2.44	32.5	0.04	2.49	2.45	33.5
	0.5	0.01	0.05	2.54	2.48	36.5	0.05	2.54	2.48	43.88	0.05	2.55	2.48	32.79	0.03	2.52	2.48	36.25
	0.5	0.1	134.78	57.58	2.5	33	127.41	50.66	**2.36**	63.38	127.38	50.7	2.5	37.82	0.04	2.53	2.48	36.75
	0.5	0.3	134.75	57.53	2.46	35.88	638.83	387.77	2.47	64.25	127.36	50.64	**2.39**	39.3	0.04	2.54	2.45	37.25
	0.1	0.001	167.04	98.99	2.41	18.75	427.95	243.47	2.45	19.5	0.05	2.58	2.51	16.25	127.35	50.67	2.4	18.13
	0.1	0.01	0.06	2.54	2.48	19	255.42	99.02	2.43	36.75	0.06	2.54	2.48	20.59	268.21	147.07	2.49	20.13
	0.1	0.1	0.36	2.67	2.45	19.88	419.91	484.4	2.49	60.13	0.34	2.65	2.45	29.11	419.96	291.56	2.46	19.63
	0.1	0.3	0.04	2.51	2.44	18.88	631.47	869.63	2.46	61	0.04	2.52	2.44	29.08	0.05	2.51	2.41	18.88
	0.01	0.001	254.75	98.87	2.46	5.75	764.52	291.55	2.45	4.63	268.31	147.09	2.49	1.88	1046.61	1265	2.58	7.25
	0.01	0.01	254.71	98.79	2.44	2	384.67	630.56	2.53	4.75	651.74	339.77	2.47	1.5	509.88	387.89	2.43	1.88
	0.01	0.1	255.22	98.99	2.41	1.75	427.94	1014.01	2.54	43.5	757.09	339.76	2.44	2.13	300.86	339.88	2.53	2.5
	0.01	0.3	166.98	98.93	2.45	5.5	607.01	629.02	2.5	47.38	536.33	291.47	2.48	5.25	509.81	195.22	2.49	1.5
B-07	$X*$		220.19	239.28	15.93	25.75	1082.15	1178.87	16.22	65.13	221.68	238.82	14.89	24.88	214.03	293.52	14.49	24.5
	1.0	0.001	308.06	168.57	2.55	32.38	732.1	279.32	2.58	32	146.41	57.93	2.56	33	0.05	2.58	2.5	31.38
	1.0	0.01	0.03	2.59	2.55	34	146.39	57.92	2.55	39.5	191.7	113.3	2.58	33.4	146.4	57.93	2.51	29.14
	1.0	0.1	191.67	113.27	2.51	32.63	146.7	58.05	2.52	63.38	1146.4	57.92	**2.47**	33.25	0.03	2.67	2.54	28
	1.0	0.3	0.06	2.56	2.49	34.75	492.13	279.48	2.5	69	146.40	57.90	2.51	35.625	439.17	279.28	2.52	35.43
	0.5	0.001	0.08	2.58	**2.46**	42.5	0.03	2.57	2.54	44.25	0.03	2.57	2.52	41.38	0.04	2.57	2.51	47
	0.5	0.01	0.04	2.59	2.53	41.63	0.06	2.58	**2.49**	50.38	0.053	2.63	2.55	44.5	0.04	2.6	2.54	34.57
	0.5	0.1	146.35	57.97	2.54	42.5	292.91	113.28	2.52	68.38	0.042	2.57	2.50	43.75	146.38	57.88	2.5	33.86
	0.5	0.3	0.03	2.56	2.51	37.38	313.09	223.94	2.51	69	0.046	2.59	2.5	41	0.317	2.7	**2.46**	36.29
	0.1	0.001	0.04	2.56	2.51	20	439.31	168.61	2.51	20.38	0.03	2.59	2.56	20	0.04	2.59	2.5	23.29
	0.1	0.01	293.07	113.43	2.51	20.88	870.04	390.05	2.54	26.5	491.98	279.47	2.51	20.5	0.16	2.67	2.53	17.29
	0.1	0.1	0.29	2.69	2.53	21	292.76	113.26	2.53	64.88	313.2	224.01	2.51	20.875	585.72	223.95	2.48	17.14
	0.1	0.3	439.27	168.66	2.56	22.63	643.45	500.99	2.57	65.75	0.026	2.60	2.55	20.875	0.05	2.56	2.49	16.86
	0.01	0.001	850.24	556.09	2.48	1.88	192	113.42	2.54	3.63	308.5	168.76	2.54	2.75	146.28	58.13	2.58	2
	0.01	0.01	443.18	224.18	2.5	3.13	691.59	556.44	2.53	5.88	0.17	2.63	2.5	2.38	586.01	224.07	2.51	2.29
	0.01	0.1	0.05	2.57	2.48	3	367.23	334.94	2.56	46.63	575.5	334.87	2.56	2.75	308.43	168.85	2.58	2.29
	0.01	0.3	575.26	334.67	2.52	2	880.28	556.87	2.52	51.75	146.38	57.98	2.51	2.38	0.18	2.67	2.57	2

Table 6: Results (mean σ and standard deviation μ) Produced by Different Combinations of Genetic Operators, Crossover Probabilities $P_c = \delta$ and Mutation Probabilities $P_m = \gamma$ for each WSRP Instance Running the SSGA For 200 Generations With a Population Size of 100.

6 Conclusion

Reducing the number of soft constraint violations is one of the challenges when applying evolutionary methods to workforce scheduling and routing problems (WSRP), which are difficult and very constrained real-world scheduling problems. Constraints in these problems include those related to workers skill qualifications, time availability and time-conflicts. In this study we investigated the performance of various generic and problem-specific genetic operators in their ability to reduce the number of constraint violations. It is apparent that not using a repair operator results in many constraint violations regardless of the combination of genetic operators used. The different combinations of genetic operators are co-dependent on the use of a repair heuristic that uses problem domain knowledge. A simple time-conflict repair heuristic (TCRH) was used here for improving the initial solutions quality. The simple Steady State Genetic Algorithm (SSGA) used here converged earlier to local minima when using TCRC than when using completely random initial solutions. However, not reducing time-conflict occurrences results in far worse outcome on the quality of solutions over the algorithm generations. Therefore, TCRH might give worse solutions in terms of cost (but with less time-conflicts) in the initial population but will help to obtain less time-conflicts in the longer run. Well-known generic operators, three mutation operators and seven crossover operators that have performed well on related problems were investigated here on their performance on the WSRP. Additionally, two cost-based genetic operators were developed with some problem domain knowledge. According to our experiments, all considered operators are suitable for the WSRP. However, some of the generic operators such as IM, PMX, HX and CX exhibited poorer performance. The SSGA provided solutions that can be improved further using problem-specific operators aided by the repair operator. In this paper we were only interested in exploring relative strengths and weaknesses of various genetic operators on the WSRP. We did not aim to design the most competitive algorithm for the problem. Consequently, we have restricted ourselves to using a simple evolutionary processes with no improvements such as local search. We believe that this study gives us guidelines for the future development of more effective algorithms incorporating genetic operators to tackle the WSRP.

References

[1] Chananes Akjiratikarl, Pisal Yenradee, and Paul R Drake. Pso-based algorithm for home care worker scheduling in the {UK}. *Computers & Industrial Engineering*, 53(4):559–583, 2007.

[2] J Arturo Castillo-Salazar and Dario Landa-Silva. A greedy heuristic for workforce scheduling and routing with time-dependent activities constraints. *Proceedings of the 4th International Conference on Operations Research and Enterprise Systems (ICORES 2015)*, pages 367–375, January 2015.

[3] J Arturo Castillo-Salazar, Dario Landa-Silva, and Rong Qu. Workforce scheduling and routing problems: literature survey and computational study. *Annals of Operations Research*, pages 1–29, 2014.

[4] Vincent A Cicirello and Robert Cernera. Profiling the distance characteristics of mutation operators for permutation-based genetic algorithms. In *The Twenty-Sixth International FLAIRS Conference*, 2013.

[5] Peter Cowling, Nic Colledge, Keshav Dahal, and Stephen Remde. The trade off between diversity and quality for multi-objective workforce scheduling. In Jens Gottlieb and GntherR. Raidl, editors, *Evolutionary Computation in Combinatorial Optimization*, volume 3906 of *Lecture Notes in Computer Science*, pages 13–24. Springer Berlin Heidelberg, 2006.

[6] Larry J Eshelman. The chc adaptive search algorithm: How to have safe search when engaging in nontraditional genetic recombination. *Foundations of genetic algorithms*, pages 265–283, 1990.

[7] David B Fogel. A parallel processing approach to a multiple travelling salesman problem using evolutionary programming. In *Proceedings of the fourth annual symposium on parallel processing*, pages 318–326, 1990.

[8] Maik Günther and Volker Nissen. A comparison of neighbourhood topologies for staff scheduling with particle swarm optimisation. In Brbel Mertsching, Marcus Hund, and Zaheer Aziz, editors, *KI 2009: Advances in Artificial Intelligence*, volume 5803 of *Lecture Notes in Computer Science*, pages 185–192. Springer Berlin Heidelberg, 2009.

[9] Sönke Hartmann. A competitive genetic algorithm for resource-constrained project scheduling. *Naval Research Logistics (NRL)*, 45(7):733–750, 1998.

[10] Ketan Kotecha, Gopi Sanghani, and Nilesh Gambhava. Genetic algorithm for airline crew scheduling problem using cost-based uniform crossover. In Suresh Manandhar, Jim Austin, Uday Desai, Yoshio Oyanagi, and AsokeK. Talukder, editors, *Applied Computing*, volume 3285 of *Lecture Notes in Computer Science*, pages 84–91. Springer Berlin Heidelberg, 2004.

[11] Wasakorn Laesanklang and Dario Landa-Silva. Mixed integer programming with decomposition to solve a workforce scheduling and routing problem. *Proceedings of the 4th International Conference on Operations Research and Enterprise Systems (ICORES 2015)*, pages 283–293, January 2015.

[12] Jan Karel Lenstra and A. H. G. Kan. Complexity of vehicle routing and scheduling problems. *Networks*, 11(2):221–227, 1981.

[13] Kevin Martin and Mike Wright. Using particle swarm optimization to determine the visit times in community nurse timetabling. In *Proceedings of the 7th International Conference on the Practice and Theory of Automated Timetabling (PATAT08), Montreal, Canada*, 2008.

[14] Melanie Mitchell. *An introduction to genetic algorithms*. The MIT Press, Cambridge, MA, USA, 1998.

[15] M Mutingi and Charles Mbohwa. Home healthcare worker scheduling: a group genetic algorithm approach. In *Proceedings of the World Congress on Engineering*, volume 1, pages 3–5. WCE, http://www. iaeng. org, 2013.

[16] I. M. Oliver, D. J. Smith, and J. R. C. Holland. A study of permutation crossover operators on the traveling salesman problem. In *Proceedings of the Second International Conference on Genetic Algorithms on Genetic algorithms and their application*, pages 224–230, Hillsdale, NJ, USA, 1987. L. Erlbaum Associates Inc., L. Erlbaum Associates Inc.

[17] Stephen Remde, Peter Cowling, Keshav Dahal, and Nic Colledge. Evolution of fitness functions to improve heuristic performance. In Vittorio Maniezzo, Roberto Battiti, and Jean-Paul Watson, editors, *Learning and Intelligent Optimization*, volume 5313 of *Lecture Notes in Computer Science*, pages 206–219. Springer Berlin Heidelberg, 2008.

[18] Franz Rothlauf. Representations for genetic and evolutionary algorithms. *Studies in Fuzziness and Soft Computing*, 104:9–32, 2003.

[19] Michael D Vose. *The simple genetic algorithm: foundations and theory*, volume 12. MIT press, Cambridge, MA, USA, 1999.

[20] D.-Z. Zheng and L Wang. An effective hybrid heuristic for flow shop scheduling. *The International Journal of Advanced Manufacturing Technology*, 21(1):38–44, 2003.

[21] Kenny Qili Zhu. A new genetic algorithm for vrptw. In *Proceedings of the International Conference on Artificial Intelligence*, page 311264. Citeseer, 2000.